02-PA-320/DP; SCH047BUS/BF/if; 2110-90-3

NON VOLATILE MEMORY DEVICE ARCHITECTURE, FOR INSTANCE A FLASH KIND, HAVING A SERIAL COMMUNICATION INTERFACE

PRIORITY CLAIM

[1] This application claims priority from European patent application No. 02425730.5, filed November 28, 2002, which is incorporated herein by reference.

TECHNICAL FIELD

[2] The present invention relates to a non volatile memory device architecture, for instance a Flash kind, having a serial communication interface, in particular for applications with Serial Protocol Interface (SPI) serial communication protocols.

[3] More specifically, but not exclusively, the invention relates to a standard Flash memory integrated in an input/output interface block to receive memory data and/or addresses from and toward the outside of the device, said interface operating according to a SPI serial communication protocol.

[4] The invention relates to the development of integrated non volatile memories having a SPI (serial Protocol Interface) serial communication interface, but the following description is made with reference to this field of application for convenience of illustration only.

BACKGROUND

[5] As it is well known in this specific technical field, a standard Flash EEPROM memory device is integrated on semiconductor with a serial communication interface equipped with an input pin INPUT PAD, an output pin OUTPUT PAD, a system clock signal pin CK_PAD and some other control pins, like SELECT_PAD.

[6] The block scheme of FIG. 1 shows the most important memory device

blocks and the SPI serial interface.

[7] Up to now the serial interface of a non volatile Flash memory has been used in the applications according to a precise communication protocol. Another parallel interface has however been used almost exclusively in the device testing phase, in order to reduce the testing time.

[8] Essentially, all the addresses and data being considered are used in the parallel mode, while only the clock signal CLK and the other four pins are necessary for the serial communication protocol.

[9] The memory device receives through the input pin all protocol codes and all information concerning the addresses, the latter being parallelised by the SPI interface and brought to the flash memory through the address bus ADDR<20:0>.

[10] Afterwards, for example in a reading operation, read data are forced by the Flash Core onto the data bus DBUS<15:0> and subsequently serialised and brought outside by means of the output pin.

[11] The system clock scans the several protocol phases according to the timing sequence shown in **FIG. 2**.

[12] As shown in this **FIG. 2**, the falling edge of the signal applied to the pin SELECT_PAD turns the memory device on and enables all input buffers, the first eight clock beats serve for the instruction code (Read, Write, etc.), the following twenty-four clock beats serve to move to the location address wished to be read or written, they follow therefore the beats required to synchronize the output data on OUTPUT_PAD.

[13] The output bit number is not previously fixed but it can be set by the user through the SELECT_PAD, the last signal rising edge indicating the end of a reading operation.

[14] At present, the testing flux software at both EWS (on-wafer-testing) and Final Test (assembled device testing) level has been conceived for a parallel-mode operation.

[15] The need to produce low-cost packages leads inevitably to the reduction of the external pin number with subsequent need for a new testing flow, resulting therefore in devices having externally only the pins required by the serial communication mode with the corresponding and necessary control pins for this mode.

[16] There are devices allowing to perform reading/writing operations on flash memories by using a system clock signal scanning the phases of a precise communication protocol at working frequencies of 25 MHz.

[17] One of the great disadvantages in the development and industrialization of this kind of devices is the cost in terms of testing time especially in the EWS phase.

[18] In fact, the whole flux serial execution involves a considerable waste of time.

[19] The technical problem underlying embodiments of the present invention is to provide a non volatile Flash memory architecture equipped with an input/output interface, particularly for SPI serial communication protocol applications, having such structural and functional characteristics as to allow a fast and low-cost testing flux operating yet in the serial mode.

SUMMARY

[20] The solution idea underlying an aspect of the present invention is to provide a sole input/output interface capable of performing both functions of the serial and parallel type, but with a reduced pin number. Essentially, the parallel operation modes are emulated by the serial interface.

[21] On the basis of this solution idea, the technical problem is solved by an architecture as previously described and defined in the characterizing part of claim 1 attached.

[22] The features and advantages of the non volatile memory architecture according to the various aspects of the present invention will be apparent from the following description of an embodiment thereof given by way of non-limiting example with reference to the attached drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[23] **FIG. 1** shows schematically a non volatile memory device architecture according to the prior art;

[24] **FIG. 2** shows schematically in an equal-time-based diagram a set of signals applied to and emitted by the memory device of **FIG. 1** in accordance with a serial testing protocol according to the prior art;

[25] **FIG. 3** shows schematically an example of a non volatile memory device architecture according to an embodiment of the invention;

[26] **FIG. 4** shows schematically on an equal-time-based diagram a set of signals applied to and emitted by the memory device of **FIG. 3** in accordance with a testing protocol according to an embodiment of the invention;

[27] **FIG. 5** is a schematic block view of the pseudo-parallel interface architecture incorporated in the device of **FIG. 3** according to an embodiment of the invention;

[28] **FIGS. 6** and **7** are respective schematic views showing fluxes of addresses and data exchanged in the interface of **FIG. 5** according to an embodiment of the invention.

DETAILED DESCRIPTION

[29] With reference to the drawings, and particularly to the examples of **FIGS. 3** and the following ones, the semiconductor-integrated non volatile memory device architecture with an input/output interface **2** is globally and schematically indicated with 1.

[30] Memory device means any monolithical electronic system incorporating a matrix of memory cells ***3***, organized in rows and columns, as well as circuit portions associated to the cell matrix and in charge of the addressing, decoding, reading, writing and erasing functions of the memory cell content.

[31] A device of this kind can be for example a semiconductor-integrated memory chip and of the non volatile EEPROM Flash type split in sectors and electrically

erasable.

[32] As it is known, each memory cell comprises a floating gate transistor having source, drain and control gate terminals.

[33] Among the circuit portions associated to the cell matrix a row decoding circuit portion is provided, which is associated to each sector and supplied with specific positive and negative voltages generated internally in the integrated memory circuit by means of positive voltage boosters or charge pumps and regulated by means of corresponding voltage regulators.

[34] Advantageously, according to an embodiment of the present invention, device *1* is equipped also with a pseudo-parallel interface *2* with a low pin number and it exploits some preexistent circuit structures in order to reduce the testing cost of the device itself.

[35] The pseudo-parallel mode here provided requires at least eleven address pins and eight data pins.

[36] In particular device *1* is equipped with all the pads necessary for a complete debut at the EWS level, while at the package level only the pads necessary for the SPI serial mode operation will be then connected. Consequently, the package of the semiconductor memory device has only eight pins.

[37] **FIG. 4** shows the timing signal sequence for data reading in this new mode.

[38] The Clock signal pin serves as synchronism signal; the cell matrix columns are stored on the falling edges of this signal and reading is started by means of an ATD (Address Transition Detection) signal.

[39] The cell matrix *3* rows are stored on the ATD signal rising edge and a new ATD pulse is started.

[40] During a Clock-Pad signal complete cycle two memory locations can be read.

[41] The block scheme of **FIG. 3** shows the architecture according to an

embodiment of the invention with the mixed SPI serial and pseudo-parallel interface **2**, receiving also the signals IC_PAD, the data DATA_PAD<7:0> and the addresses ADD_PAD<10.0>.

[42] The signal on the pin IC_PAD allows the one or the other interface to be chosen, i.e. the SPI serial or the parallel, the internal address and data buses will be forced by the one or the other communication phase interface.

[43] Eleven pins ADD_PAD<10:0> are present for the address flow in the pseudo-parallel mode, one of them serving as input pin in the serial mode.

[44] Eight data pins DATA_PAD<7:0> are present for the pseudo-parallel mode, one of them being the output pin in the SPI serial mode.

[45] If compared to the prior art, the pins CK_PAD and the pin SELECT_PAD are unchanged.

[46] The various structural blocks comprised in the pseudo-parallel interface **2** will now be analyzed in greater detail, in order to describe in greater detail the two operating modes.

[47] A first block 4 "State Machine" receives the signals applied on the pins CK_PAD and on the pin SELECT_PAD and it generates the enabling signals (ST<n:0>) for the various further circuit blocks, according to the particular timing of the protocol used.

[48] A second block **5** "Instruction Decoder" receives a signal from the less significant pin ADD_PAD<0> of the address pin group and it is in charge of decoding in the SPI serial mode the several protocol commands. This block **5** receives the information by the sole input pin ADD_PAD<0> of the address pins.

[49] A third block **6** "Addlatched<23:0>" comprises a plurality of latch registers, preferably twenty-four registers, necessary for the temporary storage of the addresses ADD_PAD<10:0>.

[50] A fourth block **7** "Datal<7:0>" comprises in turn a group of latch registers,

particularly eight registers, and it serves to store temporarily the input and output data on the pins DATA_PAD<7:0>.

[51] A fifth block ***8*** "Pulse Latching Gen" generates all the enabling signals necessary for the address and data load:

[52] This fifth block ***8*** produces a series of signals described below:

[53] SPI_EN_ADD<23:0> are the address bit latching pulses in the SPI mode; each pulse stores a sole address bit.

[54] SPI_EN_DATA<7:0> are the enabling pulses for generating serially the output datum.

[55] COL_PULSE and ROW_PULSE serves to store the ADD_PAD<10:0> content in the block ***7*** Addlatched as less significant part and as most significant part, respectively. These signals are used only in the pseudo-serial mode.

[56] DQ_PULSE serves to generate output data in the pseudo-parallel mode.

[57] **FIGS. 6** and **7** show in greater detail the address and data fluxes being exchanged in blocks ***6*** and ***7*** Addlatched and Datal.

[58] The SPI serial mode will be analyzed first.

[59] The enabling signals of the various address latches are generated on the clock signal rise front, therefore the twenty-four address bits are stored.

[60] In the parallel mode, on the contrary, two latching signals are generated: the first signal Col_Pulse, fixing the content of the addresses of Add_Pad<10:0> on the first eleven address latches, less significant part, while the second signal Row_Pulse fixes the content of the addresses Add_Pad<10:0> in the most significant address latches.

[61] Block ***9*** Enable Gen is intended to generate both the first and the following latching pulses on the basis of the logic value of signal IC deciding which of the two interfaces is to be used.

[62] As far as the output data path is concerned, in a reading operation in the two modes, generally data come from the Flash memory cell matrix, indicated in the figures with Flash Core *3*, through the DBUS<7:0> and they are stored in the various locations Data[<7:0>.

[63] Afterwards, in the SPI serial operating mode, data are transferred serially outside by means of the sole output pad and the enabling signals Spi_En_DATA<7:0>.

[64] In alternative, in the parallel mode, data are transferred outside by means of the output pins DATA_PAD<7:0>, all together in parallel on the rise front of the enabling signal DATA_PULSE.

[65] The device according to the described embodiments of the present invention, through the simple addition of a combinatory logic, allows applications on SPI buses to be available and testing phases to be performed in reasonable times reducing as much as possible production and industrialization costs.

[66] A memory device including the interface *2* may be included in a variety of different types of electronic systems, such as a computer system, personal digital assistant, cellular phone, and portable storage devices like USB drives.

[67] From the foregoing it will be appreciated that, although specific embodiments of the invention have been described herein for purposes of illustration, various modifications may be made without deviating from the spirit and scope of the invention.